# Miracle3D: Team Description Paper for RoboCup 3D Soccer Simulation Leagu

GangShuang Dai, XinYu Miu, Fan Wang

Hefei Normal University China

**Abstract.** This paper describes theMiracle3D team and the developments we did. It also describes the changes to the architecture of the Miracle3D team's structure. In addition, it also includes the researches about Dynamic Role Assignment Mechanism.

## 1 Introduction

Miracle3D is a simulation soccer 3D team establishedin 2012 and has attendedseveral competitions. Miracle 3D simulation robot soccer team participated in the state competition for the first time in 2012. In 2013, Miracle3D won the champion of Anhui Robot World Cup. In the same year, itwon the Robocup3D national first prize in China and reachedthelasteightin Robocup3D IranOpen2014. In 2014, Miracle3D won the bronze medal in Anhui Robocup3D, and in the same year, it won the fourth place in China.

There were some problems with our team code. These are the problems we are studying. We haven't yet found the good solution to the problems of precise positioning of the robot and theball (particle filter for localization), Long-Distance Kicking, andthe fastest walking. We continue to improve our code, but we have a lot of problems that can't besolved in a short time. So, we decided to refer to the base code published by UT Austin Villa. In order to speed up the development of the team, and focus on theresearch of multiagent system. Now, we use the base code released by UT Austin Villa, see https://github.com/ LARG/utaustinvilla3d, and add strategies to the code.

The rest of this paper is organized as follows. In thesecond section, the basic code of UT Austin Villa RoboCup 3D Simulation is outlined. Thethirdpartintroduces our team structure. Section 4 covers the role assignment of a team. The future is inSection 5.

## 2 Overview Team

UT Austin Villa RoboCup 3D Simulation Base Code Release is highly modular, providing us with the flexibility to modify and develop.

This version includes the following features: [1]

* Omnidirectional walking engine based on double inverted pendulum model [2]
* A skill description language for specifying parameterized skills/behaviors
* Getup (recovering after having fallen over) behaviors for all robot types

* A couple basic skills for kicking one of which uses inverse kinematics [3]
* Sample demo dribble and kick behaviors for scoring a goal
* World model and particle filter for localization
* Kalman filter for tracking objects
* All necessary parsing code for sending/receiving messages from/to the server
* Code for drawing objects in the RoboViz [4] monitor
* Communication system previously provided by drop-in player challenge 4
* An example behavior/task for optimizing a kick

## 3 Team Architecture

The low layer includes a communication module and a receiving and executing module. As the last layer of the layer structure, it is used for server communication. Its function includes two aspects: sending and receiving. As the receiver, the communication module needs to obtain information from the Server and send message to the outside world through the analytical model. As the transmitter, the robot feeds the decision back to the Server through the communication module, and then passes the parsed information to the world model again, and updates the world model according to the information from the communication module.

The skills layer is also the basic action layer, where player define basic movements and skills, such as walking, shooting, positioning, intercept, etc. The skills layer is the foundation of the entire decision- making layer and the bridge between the low layer and decision layer. However, both the analysis of message and the visual positioning will have deviations, which will affect the decision-making. Relevant algorithms need to be usedto reduce the impact of errors. Decision-makers are the brains of the people responsible for coordinating team strategies based on market conditions, making different stations, passing, dribbling and so on.

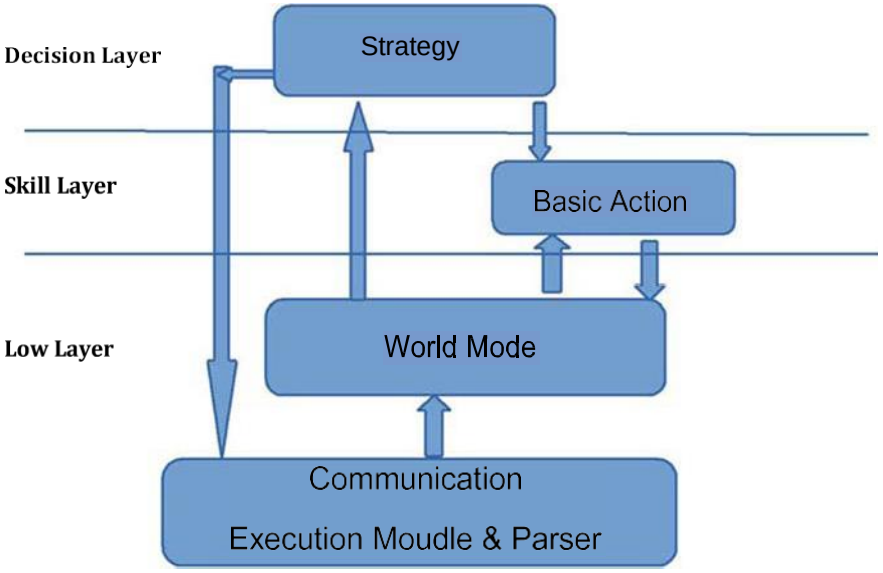


**Fig. 1.** Team Architecture of Miracle3D

# 4 Implementation of RoboCup Soccer robot Dynamic Role Assignment Mechanism

## 4.1 Strategic Design Hierarchical Model

In the design of teamwork strategy, it can be divided into three levels from top to bottom [4,5]: role allocation, behavior pattern setting and specific action setting. In this way, the coupling degree and design difficulty of each parameter are reduced effectively, and the stability and maintainability of thesystemare improved. The diagram shows the main flow diagram of the policy program implementation, where role assignment, behavior selection and data output correspond to the three levels of the model. The basic duties of all players on a football field are to attack and defend. All roles and cooperation serve both responsibilities. Based on this consideration, the basic roles of soccer robots can be divided into attackers and defenders. The attacker should have two roles of main attack and assistant attack, and the defender should have two roles of main defense and coordinated defense. Different roles have different patterns of behavior. Among the attackers, the main attack mode is to grab, dribble and shoot, assist and cover the main attack, rather than block the main attack route. The main idea of defense is to stand on the defense, stand on the dribble position and our goal line to block the attack of the other side. The difference between the main defense and the coordinated defense is that the main defense stands on a straight line precisely, and the coordinated is close to the main defense and increases the scope of defense. After determining the role and its behavior mode, according to the specific motion characteristics of the robot, the corresponding action is designed for each behavior mode, and it is converted into the speed or kick action in the robot body coordinate system and output.

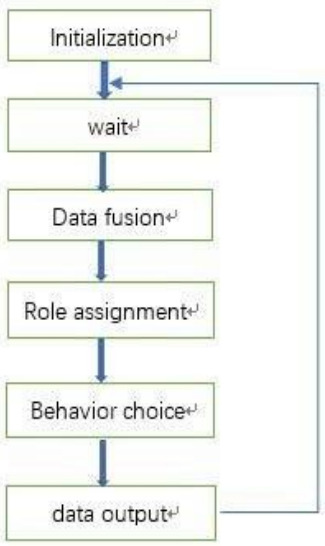


**fig 2** Hierarchical Design Process of robot Strategy

## 4.2 Mechanisms and Processes of Role Assignment

Role assignment is mainly divided into the sharing and updating of team-mate information, the improvement and sorting of team-mateinformation, the mixed allocation of the sortingsystem and priority preemption.

### 4.2.1 Information Sharing and Updating of Team Members

Assuming there are five participants, and the communication between these participants often uses multicast communication. As shown in Figure 3, each team member continuously sends some basic information to other team members and receives this information from other team members. This information needs to be shared and refined to ensure that all team members maintain the same world model. [6]

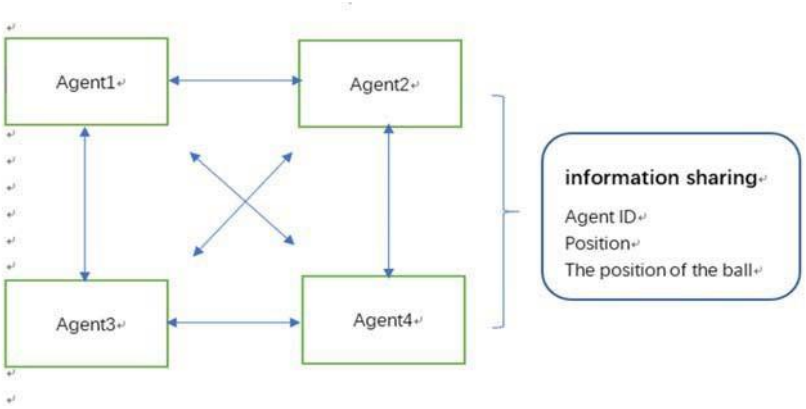


**fig 3 Diagram of communication between robots**

Global self-localization is the basic requirement of all robots' autonomous cooperation. [7] If the robot cannot achieve self- localization in the field, the above information will not be sent to the team members. For teammates, robots that can't locate themselves will be regarded as "dead in battle", and the dead robots will not participate in role assignment. "Ball-holding confidence" is a role performance function used in this link. Its value reflects the time or path cost for the robot to get the ball. Generally, the position of the robot relative to the ball is used to calculate.

The confidence function of the mechanism in actual use is as follows:

$$Q_{bc} = 1 - k_1\theta/\pi + k_2\exp(-d/3) \qquad (1)$$

Among them, K1 + K2 < 1

Qbc :The confidence value of holding the ball. d : The distance between the ball and the robot;

θ:The angle between the ball and the forward direction of the robot in the coordinate system of the robot body.

The weighting coefficients K1 and K2 represent the weights of distance and angle of confidence respectively. These two weights can be adjusted according to the motion performance of the robot. If the robot moves faster but steers slowly, it needs to increase K1 and reduce k2. On the contrary, if the robot moves slower but rotates faster, it needs to reduce K1 and increase k2. But in order to ensure that Qbc is always positive, K1 and K2 must satisfy K1+ K2 < 1. Qbc reflects the current movement and rotation costs of the ball grab robot.

In order to ensure that the role assignment of all robots is based on a unified world model, each robot needs to adjust its own world model information according to the information provided by other robot teammates. The role effectiveness function used for adjustment is to identify the similarity of the ball. If a robot cannot recognize the ball, it will directly believe that the recognition similarity of all the players is the highest; if the robot identified a ball with a lower similarity than all the players, it would choose to believe in itself or the maximum similarity based on the error of the distance function. This ensures that all the robots in the team make thesamejudgments about the position of theball.

### 4.2.2 Perfection and Ranking of Team Members'Information

As shown in Figure 4, in each robot program, a data table is needed to store the above information for all team members. This table not only contains the above information, but also adds a timestamp for each team member. Each time a team member's information is received, the program updates the original information in the table and add the last time it was received.

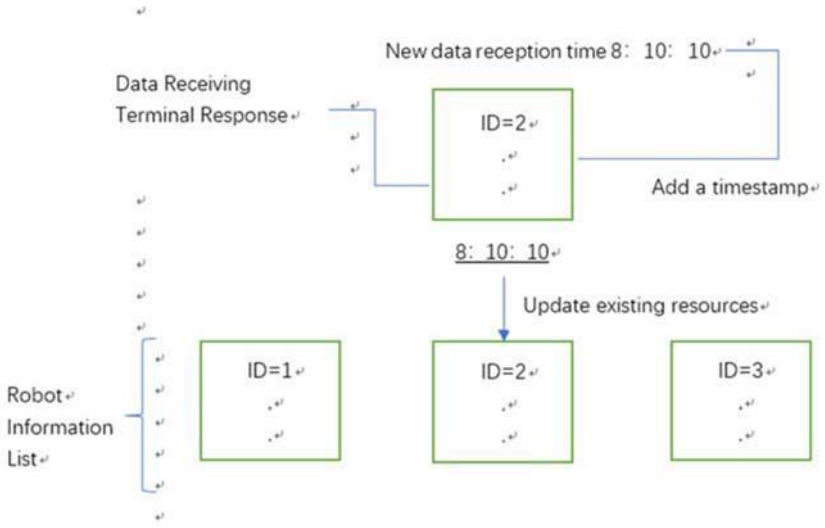


**fig 4 Add a data stamp after each receipt of team members 'data**

This data sheet is updated in real time, which means that the overtime robot is eliminated and considered dead before the roles in each main loop are assigned. A timeout is considered if the difference between the current time and the timestamp time is greater than the timeout threshold. In practical applications, the time-out threshold is generally set between 50 and 100 ms, depending on whether the network communication is smooth. All dead robots will not affect the role allocation of other robots, and when the dead robots return to normal, they will be re-added to the list of robot information.

After receiving and updating the latest team member information list, each robot performs a series of geometric and mathematical operations on the team information to find out its own grade, anduses the ranking system to seize roles and realize the role allocation of the whole team. These rankings mainly include: confidence ranking, rankingdistance from the opponent'starget,ranking distance from the target, ranking distance fromtheopponent's target. The advantage of using ranking system is that the reliability of role assignment can be guaranteed as long as the real-time update of teammate information is guaranteed.

### 4. 2. 3 Role Determination

The character decides to first judge whether it can hold confidence based on the ranking of the primary offensive ball, or it can judge whether it can serve as an assistant according to multiple conditions, or it can determine the main defense and the coordination the defense rolethrough priority preemption. Based on various rankings and teammate data, each robot makes a judgment and decides its role. The main procedures are as follows:

(1)According to the ranking of Qbc, if it is the highest, then it is the main attack object.

(2)If the above conditions do not meet, it is necessary to determine whether the conditions for assists are met. The conditions of assists depend on the specific situation. The conditions used in the actual program implementation are as follows.

$$(d_{ball}<d_{ca}) \cup (rank_{ed}>rank_{edpa}) \quad (2)$$

dball:The distance between the robot and the ball; dca:A fixed value; ranked:Ranking of goal distance from the opponent among all players;

rankedpa:Ranking of the distance between the main attack and the opponent's goal in all players.

For example, dca = 4 means it can be an assistant if the distance from the ball is less than 4m, or it can also be an assistant if the ranking is higher than the main attack.

According to the analysis of the match situation, when the assists are closer to the

opponent's goal than the main attack with the ball, or closer to the ball, the success rate of the cover cooperation will be greatly improved.

(3)If the above conditions do not meet, it is necessary to determine whether to seize the main defensive position. By calculating a point P line between our goal and the ball, we can judge the distance between the defensive position and point P and the defensive state of all teammates less than a certain range (such as the diameter of the robot), then we think they are preempted. If it is not preempted, it will be identified as its own defensive position and become the main defense.

(4) If the main defense position is preempted, the cooperative defense position will be iterated twice, and the optimal position will be selected as one of the cooperativedefense positions.

The reason for the maximum of two calculations is that there are only five robots on each side of the court, even if all of them are in defensive state. When a player occupies the main defensive position, his left and right positions become the preemptive positions of the first-level defensive players. The left and right positions of the first-level defensive players on the left will become the second-level defensive players.

The position to be occupied. The total number of main defenses, first-level and second- level co-defense places is 5, which is enough to allocate all the players on the field. The defensive players preempt the advanced defensive position in the preemption, that is to say, they preempt from primary defense to primary defense to secondary defense. Under the same priority, they will choose other better conditions (such as shorter distance, simpler path, etc.).

When the opponent is attacking with the ball, because of the change of the position of the ball, there may be a phenomenon of switching between the main defense and the co-defense. But the main defense and co-defense can always form a whole moving with the ball, so it is still effective in terms of defensive effect.

## 5 concluding remarks

In this paper, a hybrid mechanism for dynamic role assignment of soccer robots is proposed. Firstly, the role of role assignment in team cooperation strategy is elaborated, and the strategy design is divided into three levels: role assignment, behavior pattern setting and specific action setting. Then a real-time and effective role assignment mechanism is introduced, which makes full use of a small amount of communication data between robots, updates and synchronizes the status of each robot, and achieves role assignment by using multipleefficiencyfunctions, sorting system and prioritypreemptionpriority.

## 6 Future Works

In order to solve the formation and role assignment problems of football teams, it is necessary to study the dynamic strategy of the agent in its dynamic environment, and to establish a systematic robocup3d robot's action optimization mechanism. Based on reinforcement learning, the related algorithms are used to perform the agent's kicking and walking parameters Training optimization.

## 7 Acknowledgements

Now, our team is based on basic code published by UT Austin Villa. Thanks to the teammates of UT Austin Villa's teammates, they released stable base code. We thank them for their effort, publications and theories.

## 8 References

1 Patrick MacAlpine, Peter Stone. UT Austin Villa RoboCup 3D Simulation Base Code Release. In:Proceedings of the RoboCup International Symposium 2016 (RoboCup 2016),Leipzig, Germany, July 2016.

2 MacAlpine, P., Barrett, S., Urieli, D., Vu, V., Stone, P.: Design and optimization of an omnidirectional humanoid walk: A winning approach at the RoboCup 2011 3D simulation

competition. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12) (July 2012)

3 MacAlpine, P., Urieli, D., Barrett, S., Kalyanakrishnan, S., Barrera, F., Lopez-Mobilia, A., Ştiurcă, N., Vu, V., Stone, P.: UT Austin Villa 2011: A champion robot inthe RoboCup 3D soccer simulation competition. In: Proc. of 11th Int. Conf. on Autonomous robots and Multiagent Systems (AAMAS 2012) (June 2012)

4 O'Rourke J. Computational Geometry in C［M］. Beijing: Mechanics 5. Industry Press , 2005: 161— 165.

5 Bowyer A. Computing Dirichlet tessellations［J］. The Computer Journal, 1981, 24( 2) : 162—166.

6 Amenta N, Bern M, Kamvysselis M. A new Voronoi —basedsurface reconstruction algorithm ［C］/ / Proceeding of SIGGRAPH'98. Danvers: Assison—Wssley Publishing Company, 1992: 415—421.

7 Edelsbrunner H, Shah N R. Incremental topological flipping works for regular triangulations［J］. Algorithmica, 1996, 15( 3) : 223—241.

8 Rongyi He,Chunguang Li. RoboCup3D simulation robot gait optimization research [J]. Computer and Modernization, 2018 (3).